CPU-less robotics: distributed control of biomorphs

Ralph Etienne-Cummings, M. Anthony Lewis, Mitra Hartmann, Avis H. Cohen

Ralph Etienne-Cummings, M. Anthony Lewis, Mitra Hartmann, Avis H. Cohen, "CPU-less robotics: distributed control of biomorphs," Proc. SPIE 4109, Critical Technologies for the Future of Computing, (17 November 2000); doi: 10.1117/12.409207

Event: International Symposium on Optical Science and Technology, 2000, San Diego, CA, United States

CPU-Less Robotics: Distributed Control of Biomorphs

Ralph Etienne-Cummings[a,b], M. Anthony Lewis[b], Mitra Hartmann[c] and Avis H. Cohen[d]

[a]Dept of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218

[b]Iguana Robotics, Inc., P.O. Box 628, Mahomet, IL 61853

[c]Division of Biology, California Institute of Technology, Pasadena, CA 91125

[d]Dept. of Biology, University of Maryland, College Park MD 20742

ABSTRACT

Traditional robotics revolves around the microprocessor. All well-known demonstrations of sensory guided motor control, such as jugglers and mobile robots, require at least one CPU. Recently, the availability of fast CPUs have made real-time sensory-motor control possible, however, problems with high power consumption and lack of autonomy still remain. In fact, the best examples of real-time robotics are usually tethered or require large batteries.

We present a new paradigm for robotics control that uses no explicit CPU. We use computational sensors that are directly interfaced with adaptive actuation units. The units perform motor control and have learning capabilities. This architecture distributes computation over the entire body of the robot, in every sensor and actuator. Clearly, this is similar to biological sensory-motor systems. Some researchers have tried to model the latter in software, again using CPUs.

We demonstrate this idea in with an adaptive locomotion controller chip. The locomotory controller for walking, running, swimming and flying animals is based on a Central Pattern Generator (CPG). CPGs are modeled as systems of coupled non-linear oscillators that control muscles responsible for movement. Here we describe an adaptive CPG model, implemented in a custom VLSI chip, which is used to control an under-actuated and asymmetric robotic leg.

Keywords: Central Pattern Generation, Biomorphic Robots, VLSI Locomotion Controller, Legged Locomotion

1. INTRODUCTION

1.1 Motivation

Challenges for robotics in the future include the miniaturization of walking, running, and flying robots, increasing the real-time adaptability of robots to the environment, and the creation of mass-market consumer devices (e.g. Sony Dog [1]). These new technologies will require small, low-cost, power-efficient, and adaptive controllers, and may therefore benefit greatly from computational support, i.e. neuromorphic engineering, that is radically different than current microprocessor-based technology.

The current state-of-the-art of biologically inspired robotics, represented by the Honda's humanoid robot or Sony's Dog, depends heavily on microprocessors for all computation. Consequently, the availability of embedded high-performance CPUs has only recently allowed the development of these un-tethered and "autonomous" robots. Despite their supposed biomimetic nature, the computational mechanisms employed in either of these robots are decisively non-biological. To our knowledge, CPUs do not model any neural circuits nor do they employ any biological control strategies and architectures, other the external appearance.

Does bio-mimicry require that every single details of the biological master be incorporated in the engineered system? The answer is resoundingly "no." Information processing in the brain can be modeled as many levels. Traditionally, neuromorphic engineering deals with reverse engineering a small subset of the details. An example is the silicon neuron of Mahowald and Douglas [2]. We choose to model these system at a functional level which is more amenable to an engineering solution. In our paradigm, the 'brain' of our robots is modeled as a number of interconnected sub-processors. Each sub-processor roughly reflects the functionality embedded in a brain region or aggregate of brain regions. Furthermore, we require that each subsystem is realizable with compact, low-power Very Large Scale Integrated (VLSI) circuits. The latter requirement reflects the reality of developing autonomous robots that can be small, light-weight, fast and power efficient. This is in stark contrast to the Honda humanoid robot.

Our main thesis is that distributed processing with autonomous computational sub-processors that are adaptive and responsive to local sensor information is necessary in real-time robotics that interact with their environments. Local and distributed learning capability is a fundamental requirement of such systems. It allows the robot to adapt to variations in the electronics, mechanics and environment. Secondly, these sub-processors are naturally realized in custom VLSI circuits. The resulting advantage will be a combination of highly robust, intelligent behavior, implemented in compact, low power circuits.

To realize our system, we adopt the neuromorphic engineering philosophy of Carver Mead [3]. The basic principle of neuromorphic engineering is to use principles of biological information processing to address real-world problems. Using a neuromorphic approach, complete nervous systems can be built to control robots. These artificial nervous systems can be realized in very low cost, low power and low weight units. It is well recognized that the physics of silicon is in many ways analogous to the biophysics of the nervous system [3]. Therefore, neuromorphic systems are often implemented in silicon using as much of the properties of device physics as possible. However the vast majority of work in neuromorphic engineering to date has concentrated on sensory processing (for example, the construction of silicon retinas [4] or silicon cochleas [5]).

Iguana Robotics, Inc., and Johns Hopkins University are developing next generation visually guided walking machines. These walkers will be more robust, smarter, faster and have more elegant movement than the current state-of-the-art. They will handle rugged environments, *learn* to jump over 3D obstacle and smoothly change gaits from walking to running to jumping based on *passive* computational vision. The environment will be sensed using a custom designed compact vision systems, realized primarily in VLSI. Hardware models of neural sub-circuits in motor nuclei, spinal cord, lower brain, cerebellum and visual cortex of vertebrates will be used to control the robot. Because legged locomotion is one of the hardest capability to realize in biomorphic robots, it has been tackled first. We have developed one of the first adaptive Central Pattern Generation (CPG) chips and applied it to control a pair of running legs.

In this paper we present a chip, based on established principles of the locomotor-control circuits in the nervous system, that mimics many of the features of a biological Central Pattern Generator (CPG). We show that the circuit, consuming less than one microwatt of power and occupying less than 0.4 square millimeters of chip area can generate the basic competence needed to control a robotics leg running on a circular treadmill. Furthermore, the circuit can use sensory feedback to stabilize the rhythmic movements of the leg.

Potentially, this technology could provide inexpensive circuits that are adaptable, controllable and able to generate complex, coordinated movements. Such circuits could be used in miniature systems to modulate repetitive cyclical movements based on appropriate sensory feedback. These systems could include miniature walking, running, flapping and swimming machines. Figure 1 shows a picture of CPG chip, adaptive neural circuits and the pair of legs used in the experiments.

2. CPG THEORY

The basic notion of an autonomous neural circuit generating sustained oscillations needed for locomotion was first articulated in the early part of this century [6]. The key idea is that an autonomous system of neurons can generate a rhythmic pattern of neuronal discharge that can drive muscles in a fashion similar to that seen during normal locomotion. Locomotor CPGs are autonomous in that they can operate without input from higher centers or from sensors. Under normal conditions, however, these CPGs make extensive use of sensory feedback from the muscles and skin, as well as descending input [9]. Furthermore, the CPG transmits information upward to modulate higher centers as well as to the periphery to modulate incoming sensory information.

The CPG is most often thought of as a collection of distributed elements. For example, in the lamprey (a relatively simple fish-like animal) small, isolated portions of the spinal cord can generate sustained oscillations. When the spinal cord is intact, these small elements coordinate their patterns of activity with their neighbors and over long distances ([7][11]).

It is well known that sensory input can modulate the activity of CPGs. Modulation of the CPG by sensory input can be seen quite clearly in the resetting of the phase of the CPG. For example, as a walking cat pushes its leg back, sensors in the leg muscles detect stretching. These sensors (called stretch receptors) signal this stretch to the nervous system. Their firing initiates the next phase of the CPG causing the leg to transition from stance to swing phase.

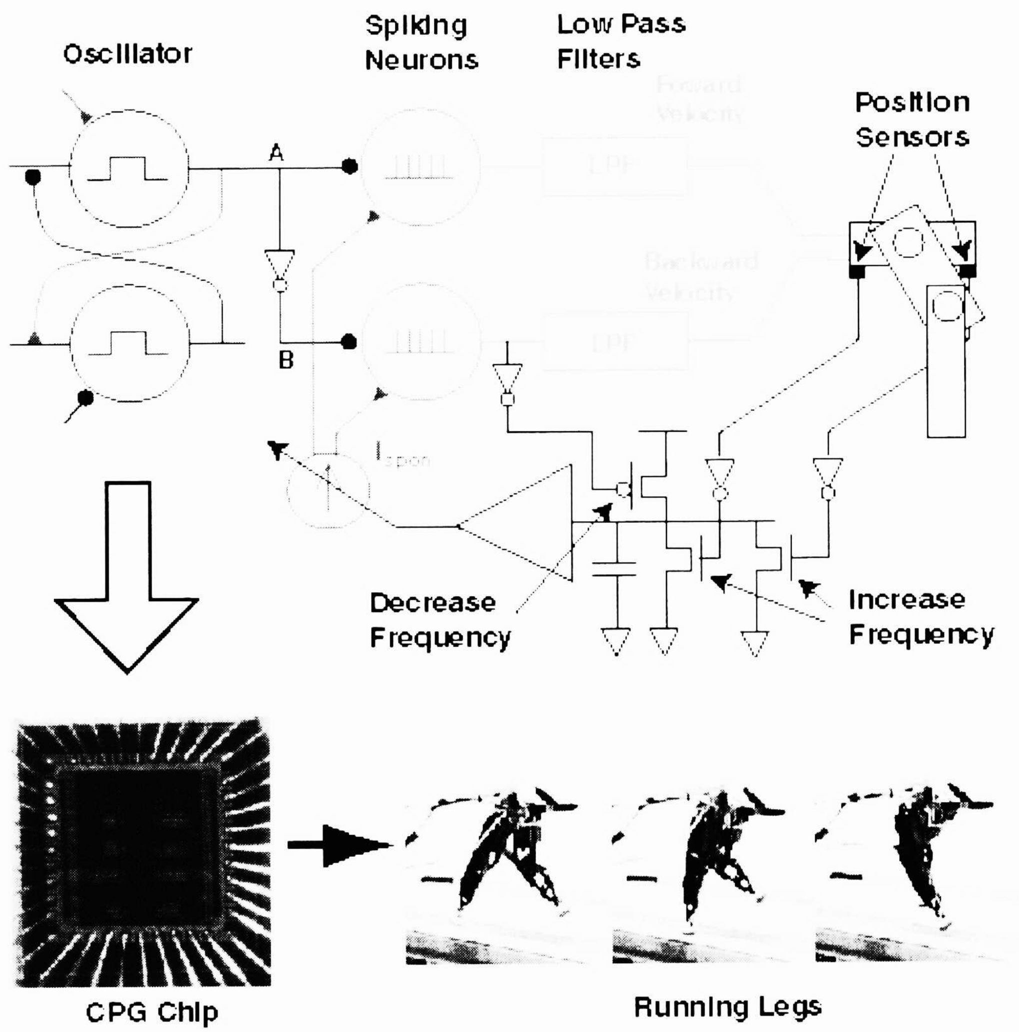

Figure 1: Using the CPG chip to adaptively control a pair of running legs.

In the early 1980s Cohen and colleagues [8] introduced a model of the lamprey CPG using a system of phase-coupled oscillators. Later, a model called Adaptive Ring Rules (ARR), based on ideas in this and related work was extended for use in robot control [12][13].

A full exposition of ARR is beyond the scope of this paper. Briefly, however, an ARR is a model of a non-linear oscillator at a behavioral level. This model is complex enough to drive a robot while also allowing easier implementation of learning rules. ARR theory inspired the philosophy behind the design of this chip.

2.1 Modeling CPGs on a Neuromorphic Chip

CPGs are most often modeled as distributed systems of non-linear oscillators. In our implementation the basic coordination in the leg is achieved by phasically coupling two neurons together to achieve oscillations. When coupled together they are alternatively active. This alternating activity is the basic coordination needed to drive the hip of the robot. A phase control circuit governs the phase difference between the neurons.

These oscillator neurons drive two integrate-and-fire spiking *motoneurons.* These neurons are used to drive an actuator. The spiking neuron could also drive biological muscle or it could also be used to drive a pneumatic cylinder, a McKibben actuator or biomuscle directly.

In our experimental setup, the robot under control uses servomotors. To be compatible with this technology, it was necessary to low-pass filter the spiking neurons and then integrate the resulting smooth graded velocity signal.

We will show the circuit in autonomous operation and with sensory feedback from stretch receptors used to reset the CPG. We also demonstrate a property of our biomorphic leg: we show that our limb and its control circuit can not only produce stable rhythmic motion, but it can also compensate for intentional biases in the chip as well as mechanical complexity of an active hip and passive knee.

2.2 Previous Work

CPG chips and circuits have been created before. For example, Still reports on a VLSI implementation similar to a CPG circuit used to drive a small robot in [17][18]. This circuit captured some of the basic ideas of a CPG but did not incorporate a motoneuron output stage, and the system did not provide for adaptation via sensory input. However, she did demonstrate rudimentary control of a walking machine.

The work of DeWeerth and colleagues [14] captures the neural dynamics on a much more detailed level than has been achieved here. There are great difficulties in applying such a system to the control of a robot. Primarily, parameter sensitivity makes such circuits difficult to tune. To address this issue, DeWeerth et al. have implemented neurons that self-adapt their firing-rate [16]. The adaptation, however, is independent of external inputs from sensors. While detailed neural models are difficult to work with in silicon, we will undoubtedly learn a great deal from these efforts in the future.

Ryckebusch and colleagues [15] created a VLSI CPG chip based on observations in the thoracic circuits controlling locomotion in locusts. The resulting VLSI chip was used as a fast simulation tool to explore understanding of the biological system. Their system did not use feedback from sensors, nor was it connected to a robotic system. However, again their objective, of modeling a particular biological circuit, was different than the objective described in this paper.

Our work differs from the previous work in several respects. First, we allow adaptation based on sensory input. Adaptation is shown as a phase resetting of the CPG based on certain sensory triggers. Firing frequency is also adapted by sensory feedback. Second, our chip has short-term memory devices that allow adaptation of the output parameters. In addition, we make use of integrate-and-fire neurons for the output motoneurons. Our abstraction is at a higher level than other reported work ([14][16]). We believe that by using a higher level of abstraction we will be able to more easily implement on-chip learning. In systems based on numerous inter-related parameters, it is not apparent how learning at the level of behavior can be coupled to low level parameter changes.

3. THE CPG CHIP

The CPG chip is designed to provide biologically plausible circuits for controlling motor systems. The chip contains electronic analogues of biological neurons, synapses and time-constants. In addition, the chip also contains dynamic analog memories, and phase modulators. Using these components, non-linear oscillators, based on the central pattern generators of biological organisms, can be constructed.

The dynamical properties of the neural circuits can also be adapted using direct sensory information. In this first version of the chip, all the components are individually accessible such that they can be connected with off-chip wiring to realize any desired circuit. In future versions, tested neural CPG circuits will be integrated with completely hardwired or programmable circuits.

3.1 The Hardware Components

3.1.1 The Neuron

Our neurons use an integrate-and-fire model. A capacitor, representing the membrane capacitance of biological neurons, integrates impinging charge. When the "membrane-potential" exceeds the threshold of a hysteretic comparator, the neuron outputs high. This logic high triggers a strong discharge current that resets the membrane potential to below the threshold of the comparator, thus causing the neuron output to reset. This circuit therefore emulates the slow phase and fast phase dynamics of real neurons. The process then starts anew. Fig. 2 shows a schematic of the neuron circuit.

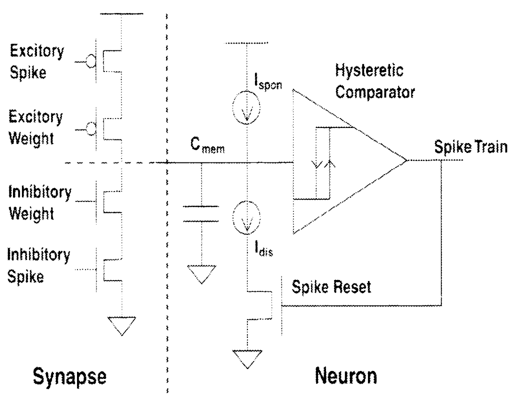

Figure 2. Schematic of the integrate-and-fire motoneuron and synapse.

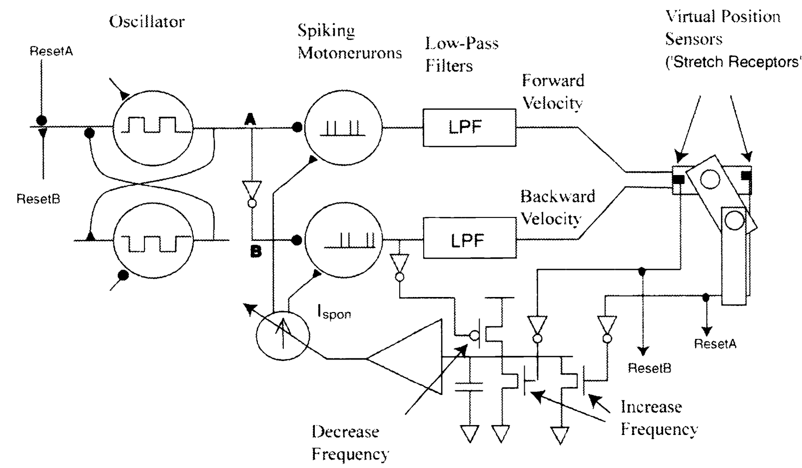

Figure 3: Adaptive control of a limb's dynamics using a neural CPG with learning capabilities.

Our neurons carry activation information in the frequency of spikes. The rate at which the membrane potential charges up controls the firing frequency of the neuron. This rate is given by the sum of the total charge flowing in and out of the membrane capacitance. The strength of the reset current source determines the width of each neural spike. The discharge current is usually set to a large value so that each spike is narrow and is not influenced by the charge injected onto the membrane capacitor. Typically, the neuron is set such that it fires at a nominal rate at rest; additional inputs increase or decrease the firing rate. Shunting inhibition can also silence the neuron.

Equation 1 gives the dynamic equation for the neuron. There are three input voltages: (1) a feedback input from a hysteretic comparator (S_{dis}), (2) Excitatory inputs from other neurons (S_i) and (3) Inhibitory Inputs from other neurons ($\bar{S}_i$). These inputs are weighted by current sources. These current sources are denoted I_{dis}, I_i and $\bar{I}_i$ respectively. In addition, a constant current injection, sets a spontaneous spike rate of the neuron. As noted above, I_{dis} sets the spike duration. Finally, the term V_T^+ and V_T^- set the thresholds for the hysteretic comparator respectively.

The spike trains impinging on a neuron activate switches that allow charge quanta to flow into or off the membrane capacitor. The amount of charge transferred per spike is the synaptic weight and is controlled by an applied voltage that regulates the current sources. Modulation of this voltage allows the adaptation of the neural firing rate and is used during learning. The left-hand side of Fig. 2 shows the schematic of the synapse, while equation 1 shows how the neuron is affected by the synaptic weight.

$$
\begin{aligned}
&C_i^{mem}\frac{dV_i^{mem}}{dt} = I_{spon} - S_i I_{dis} + \sum_j \bar{S}_j I_j^{+} - \sum_j S_j I_j^{-} && (a) \\
&S_i = \begin{cases} 1 & if \quad V_i^{mem} > V_T^{+} \\ 0 & if \quad V_i^{mem} < V_T^{-} \end{cases} && (b)
\end{aligned}
\qquad (1)
$$

In addition to spiking neurons, we make use of neurons with graded response. These neurons are essentially the same as the spiking neuron except that the hysteretic comparator is replaced with a linear amplifier stage and no feedback voltage is used.

3.1.2 The Oscillator

The neural circuits for creating the CPG are constructed using a cross-coupled square-wave oscillators. The outputs of these oscillators drive the bursting motoneurons. A master-slave configuration of the neurons allows us to construct an oscillator with a constant phase relationship. By setting the excitatory and inhibitory weights to equal values, a square-wave with a duty-cycle of 50% is obtained. The phase relationship between the two sides can be varied. The frequency of oscillation is set by the magnitude of the weights. This asymmetrically cross-coupled oscillator serves as the basic CPG unit that can be modified according to the application. By injecting or removing charge from the membrane capacitors of the oscillator neurons, the properties of the CPG can be altered.

For more complex waveforms a phase controller is included on chip. This phase controller allows the phase difference between oscillators to be set arbitrarily. For the experiments described in this paper, a strict 180 degrees phase relationship is required. Hence an inverted version of one of the oscillators is used, as shown in Fig. 3.

3.1.3 The Neural Circuit

The complete neural circuit is given in Fig 3. The output of the basic Oscillator unit is used to inhibit the firing of the spiking motoneuron. When the oscillator output is high, the motoneuron is not allowed to fire. This produces two streams of 180 degrees out of phase spike trains. These trains can be low-pass filtered to get a voltage which can be interpreted as a motor velocity. Consequently, the oscillator controls the length of the motor spike train, while the spike frequency indicates the motor velocity.

The spike frequency is regulated by a feedback loop. Spiking places charges on the neuron membrane capacitor. The integrated charges are buffered and then used to down regulate spike frequency. In this way spike frequency is less sensitive to component variations.

3.2 Sensory Adaptation and Learning

3.2.1 Adaptation based 'stretch receptor'

As shown in Fig. 3, the oscillator neurons can be stopped or started with direct inhibitory and excitatory sensory inputs, respectively. When the inputs are received as strong inhibition, the membrane capacitor will be shunted and discharged completely. It will remain in this state until the inhibition is released, then the normal dynamics of the oscillator will continue from the inactive state. On the other hand, if the sensory input is received as a strong excitation, the oscillator will be driven into an active state. When the excitation is released, the oscillator will continue from the active state. Clearly, the charge-up or discharge of the membrane capacitor will be influenced by any direct sensory input. If the sensory inputs are periodic, the oscillator outputs can be driven such that they are phase locked to the inputs. Thus the oscillator is entrained to the dynamics of the system under control.

We use this property to mimic the effect of the stretch reflex in animals. When the leg of an animal is moved to an extreme position, a special sensor called a stretch receptor sends a signal to the animals CPG causing a phase resetting. This is mimicked in the circuit presented here. Referring to Fig. 3, the leg may reach an extreme position and while still being driven by the oscillator. In this case, a virtual position sensor, which mimics a stretch receptor, sends a signal to *ResetA* or *ResetB* to cause a resetting of the oscillator circuit as is appropriate to cause a hip joint velocity reversal.

3.2.2 Spike Frequency Adaptation

If learning is required, the chip provides a short-term (on the order of seconds) analog memory to store a learned weight. Clearly, this architecture favors a continuous learning rule. Spikes from the motoneurons are used to increase or decrease a voltage on a capacitor; the voltage is used to set the connection weight of another neuron. In the absence of the training inputs, the stored weights decay at approximately 0.1V/s. Fig. 3 shows a schematic for adapting the spiking frequency of the motoneurons based on the swing amplitude of the limb.

In Fig. 3, the limb is driven back and forth with a velocity signal that is obtained by low-pass filtering the activity of the motoneurons. Since the CPG oscillator fixes the duration of the spike train, changing the spiking frequency of the motoneuron alters the amplitude of the velocity signals, which in turn varies the swing amplitude of the limb. If the amplitude of swing does not reach the maximum positions, the motoneuron spike rate is increased. An increase in spike rate is kept bounded by negative feedback to the learning circuit. When the swing amplitude reaches maximum, the positive input to the learning circuit is reduced, thus allowing the spiking rate to settle to a constant value. The continuous negative feedback of the spike rate and the input from the position detectors maintain the learned spiking rate. The duration of the burst component of the spike train can be further controlled by feeding the position signals directly to the CPG oscillators to reverse the trajectory of motion at the end points. This allows very asymmetric forward and backward velocity signals to be adaptively re-centered.

4. ADAPTIVE RUNNING

The experimental setup consists of a pair of robotic legs, the CPG chip, necessary components to interface the chip to the robotic leg, and data collection facility.

The robotic legs are small (10-cm height) two-joint mechanism. In our setup, only the "hip" is driven. The "knee" is completely passive. The knee swings freely, rotating on a low friction ball-bearing joint. A hard mechanical stop prevents the knee from hyperextending.

The neurons of the CPG chip are interfaced to a servomotor using a rudimentary muscle model. The muscle dynamics are simulated as a low pass filter to smooth the output of the spiking neurons. This is followed by integrator, implemented in software, to convert the velocity signal to a position command needed by the servomotor. A bias was intentionally introduced into the chip to cause an asymmetry in the backward and forward swing of the leg. This bias might be typical of uncompensated parameters in a chip.

Each robotic leg has three sensors on it. Two LVDT sensors monitor the position of the knee and hip joints. LVDT sensors are used because they introduced minimal friction and had infinite resolution. Additionally, the robot has miniature load-cell sensor that monitors ground forces. The units of the load cell are uncalibrated in all figures.

An oscillator frequency was selected by hand to be approximately 2-3 Hz. This frequency would excite the mechanical structure and cause the legs to "run" on a rotating drum, treadmill or on the ground. The data presented was measured for a single leg on a rotating drum.

4.1 Running with a passive knee

In this experimental setup, the CPG circuit drives the actuator in the hip joint. The knee joint is passive and rotates with very little friction. The assembly is suspended above a rotating drum. The CPG circuit is started. Data is collected for three sensors: Foot pressure, knee and hip. "Stretch receptor" sensory feedback from the hip is used as feedback to the CPG.

A remarkable feature of this system is that the knee joint adapts the correct dynamics to enable running (!). As the upper limb swings forward, the lower limb rotates so that the foot comes off the ground. When the upper limb is suddenly accelerated backward, the momentum in the lower limb forces the knee to lock in place. At just the correct moment, the foot contacts the ground and the subsequent loading keeps the knee joint locked in place. As the foot travels backward it eventually begins to unload. Stored energy in the elastic foot causes it to 'kick up' and smartly snap off the ground, an effect most noticeable at higher velocities.

Figure 4 shows a phase plot of the knee, foot and hip position and foot contact. The bulk of the trajectory describes a tight 'spinning top' shaped trajectory while the few outlying trajectories are caused by disturbances. After a disturbance the trajectory quickly returns to its nominal orbit and we can infer that that the system is stable.

4.2 Sensory feedback lesioning

This experimental setup is similar to the first experiment. The difference is that sensor feedback is lesioned (turned off) periodically. Next we lesioned the sensory feedback to the leg periodically. Figure 5 shows the effect of lesioning on the position of the hip and knee joints as well as the tactile input to the foot. After lesioning the leg drifts backward significantly due to a bias built into the chip. When the sensory input is restored, the leg returns to a stable gait.

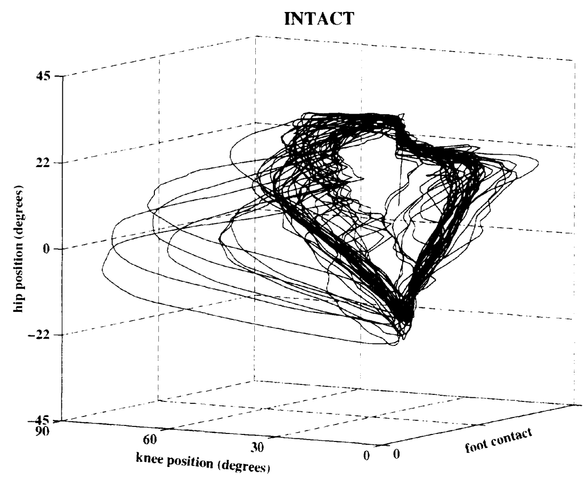

Figure 4. Hip, knee and foot-contact phase diagram. Most of the trajectory is in a tight bundle, while the outlying trajectories represent perturbations.

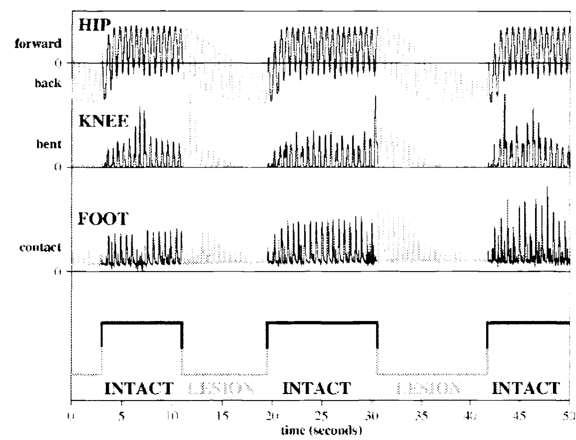

Figure 5. This figure shows the effect of lesioning sensory feedback. When the feedback is lesioned (Time 11-19 seconds and 31-42 seconds), the hip drives backward significantly. As it does the foot begins to lose contact with surface and the knee stops moving. When the lesion is reversed at 19 and 42 seconds, the regularity of the gait is restored.

5. SUMMARY AND CONCLUSIONS

In this paper we have presented the first experimental results of an adaptive aVLSI neural chip controlling a robotic leg. Using sensory feedback, the circuit can adapt the gait of the leg to compensate both for chip bias and for environmental perturbations. This work represents the first experimental results of an adaptive VLSI neural chip controlling a robot leg.

Basic rhythmic movements in animals are generated by a network of neurons in the spinal cord called the Central Pattern Generator or CPG. CPGs have been studied extensively and are beginning to be better understood. A model of the CPG was proposed by Cohen in the early 1980s and subsequently this CPG model was then adapted for use in robotic work [12][13]. In this paper we present a hardware implementation of this CPG model. Our custom VLSI chip, having only 4 neurons and occupying less than 0.4 square mm has the basic features needed to control a leg running on a treadmill.

We conclude that the control of a running leg using an aVLSI CPG chip is possible. We demonstrate that, at least in this experimental setup, running is possible using an under-actuated leg. Finally, we demonstrate a basic adaptive property of phase resetting using a stretch receptor.

Because (1) we do not make use of models, or linearization, (2) we adapt principles from biological systems, and (3) these principles can easily be implemented with low-power integrated circuits, we are able to achieve a very compact solution. Further experimentation with this system will allow us to determine if a robot can be made to walk by coupling together mulitple circuits as presented here. The current results, however, are promising.

ACKNOWLEDGEMENTS

The authors acknowledge support of Grant No. N00014-99-0984 from ONR to Lewis & Etienne-Cummings, NSF Career Grant #9896362 to Etienne-Cummings and NIH grant MH44809 to Cohen.

REFERENCES

[1] http://www.world.sony.com/robot/index.html
[2] Mahowald, M. and Douglas, R. (1995), "Silicon Neuron," *Science*, Vol. 269, No. 5226, pp. 981.
[3] Mead, C., (Ed.) (1989). *Analog VLSI and Neural Systems*, Addison-Wesley, Reading, MA.
[4] Koch, C., and Li, H., (Eds.) (1995). Vision Chips: Implementing Vision Algorithms with Analog VLSI Circuits, IEEE Computer Press.
[5] Lazzaro, J., Wawrzynek, J., and Karmer, A. (1994), Systems Technologies for Silicon Audition, In: *IEEE Micro*, Vol 14, No. 3, pp. 7-15.
[6] Brown,T.G. (1914). On the nature of the fundamental activity of the nervous centres; together with an analysis of the conditioning of the rhythmic activity in progression, and a theory of the evolution of function in the nervous system. J. Physiol., Vol. 48, pp. 18-46.
[7] Cohen, A. H. and Wallén, P. (1980). The neuronal correlate to locomotion in fish: "Fictive swimming" induced in an in vitro preparation of the lamprey spinal cord. Exp. Brain Res. Vol. 41, pp. 11-18.
[8] Cohen, A. H., Holmes, P. J. and Rand, R. H. (1982). The nature of the coupling between segmental oscillators of the lamprey spinal generator for locomotion: A mathematical model. J. Math. Biol. Vol. 13, pp. 345-369.
[9] Cohen, A. H., Rossignol, S. and Grillner, S. (1988). *Neural Control of Rhythmic Movements in Vertebrates*.Wiley & Sons.
[10] Full, R. J. (1993). Integration of individual leg dynamics with whole body movement in arthropod locomotion, In. R. D. Beer, R. E. Ritzmann, and T. McKenna, Eds, *Biological Neural Networks in Invertebrate Neuroethology and Robotics,* Academic Press, Inc, Boston.
[11] Grillner, S. and Wallén, P. (1985). Central pattern generators for locomotion, with special reference to vertebrates. Ann. Rev. Neurosci. Vol. 8, pp. 233-61.
[12] Lewis, M.A.(1999). Distributed Architecture for Gait Adaptation in a Quadrupedal Robot, *Submitted.*
[13] Lewis, M.A. (1996). *Self-organization of Locomotory Controllers in Robots and Animals,* Ph.D. Dissertation, Department of Electrical Engineering, University of Southern California, Los Angeles.
[14] Patel, G., Holleman, J. and DeWeerth, S. (1998). Analog VLSI Model of Intersegmental Coordination with Nearest-Neighbor Coupling. In Adv. Neural Information Processing, Vol. 10, pp. 791-725.
[15] Ryckebusch, S. , Wehr, M. and Laurent, G. (1994). Distinct Rhythmic Locomotor Patterns Can be Generated by a Simple Adaptive Neural Circuit: Biology, Simulation and VLSI Implementation, *J. of Comp. Neuro,* Vol 1, pp 339-358, 1994.
[16] Simoni, M., and DeWeerth, S. (1999). Adaptation in a VLSI Model of a Neuron. In: Trans. Circuits and Systems II, Vol. 46, No. 7, pp. 967-970.
[17] Still, S.(1998). Presentation at Neurobots Worshop, *NIPS*98*, Breckenridge, CO, USA.
[18] Still, S. and Tilden, M.W. (1998). Controller for a four legged walking machine, In: Neuromorphic Systems: Engineering Silicon from Neurobiology, Eds: L. S. Smith and A. Hamilton, World Scientific: Singapore, pp 138-148.